ABN-182

PCI-X Gigabit Ethernet controllers

User's Guide

Portwell Inc.
3F, No. 92, Sec. 1, Nei-Hu Rd., Taipei 114, Taiwan, R.O.C.
Headquarter: +886-2-2799-2020 FAX: +886-2-2799-1010
http://www.portwell.com.tw

About This Guide

Purpose

This manual provides information on **ABN-182** PCI-X Gigabit Ethernet card, including software installation

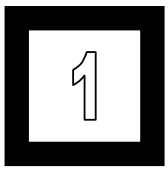
Manual Contents

Chapter 1 Overview of **ABN-182** Ethernet controllers.

Chapter 2 **ABN-182** Ethernet product detail.

Chapter 3 **ABN-182** Ethernet product programming setup detail.

Chapter 4 Driver installation detail of **ABN-182**.



Introduction

This user manual is used for ABN-182 standard universal PCI-X Ethernet controllers. The basic design of both Ethernet controller cards is identically. ABN-182 has Dual Ethernet ports and universal 64-bit standard PCI-X To enhance Ethernet performance. It designed with Intel 82546GB Gigabit Ethernet Controllers to provide two Gigabit Ethernet port. The further detail of both products will be covered in the following chapters.

Dual 64-bit Gigabit Ethernet ports base on Intel® 82546GB controller. Supports 133MHz PCI-X bus and backwards compatible with 64/32-bit Built-in Watchdog Timer (WDT) to switch Ethernet ports to bypass mode by system hang and power failure. Easy configuration of Normal / Bypass mode and WDT time-out per software programming

Built with both on-board LED for LAN status and bypass mode.

1.1 Features

- Gigabit Ethernet card with 2nd generation By-Pass function.
- Dual 64-bit Gigabit Ethernet ports base on Intel® 82546GB controller.
- 64bit 133MHz PCI-X interface.
- Built-in Watchdog Timer (WDT) function.
- Easy configuration for by-pass mode via software programming.
- Built with both on-board LED for LAN status and bypass mode,

1.2 Bypass Block Diagram

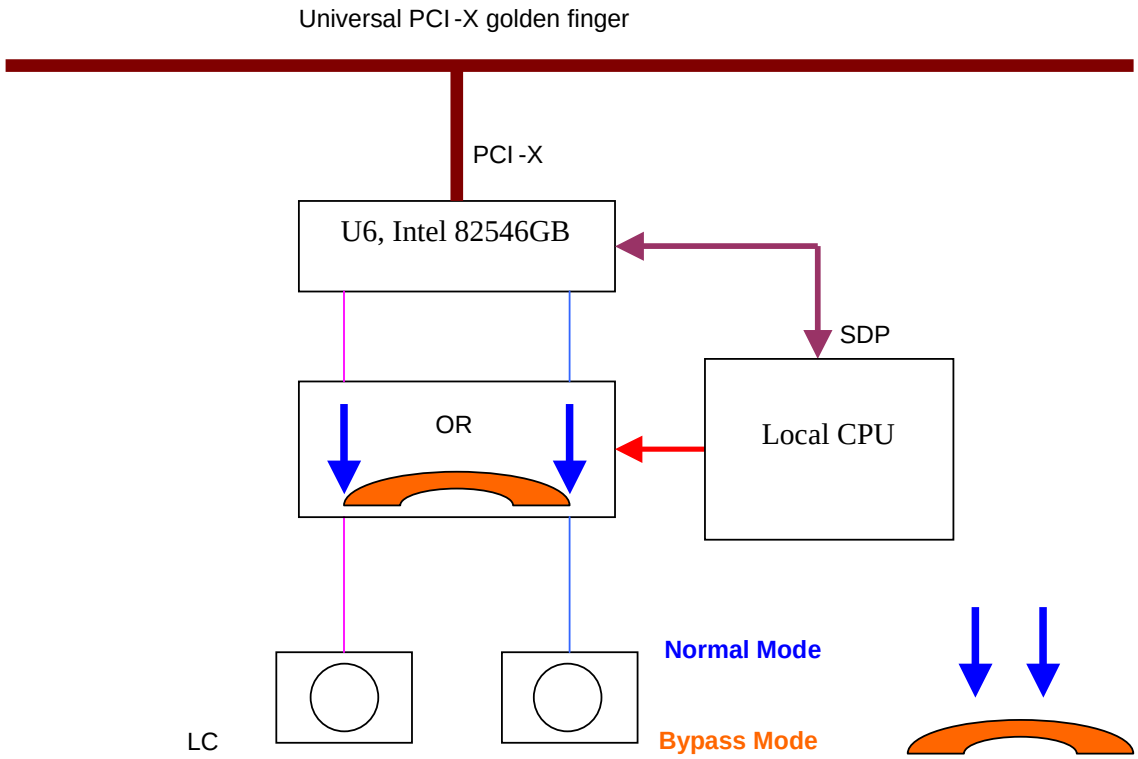


Figure 1: ABN-182 Block Diagram

2

ABN-182 PCI-X Ethernet card

2.1 Product Specification

- Gigabit Ethernet card with 2nd generation By-Pass function.
- Dual 64-bit Gigabit Ethernet ports base on Intel® 82546 GB controller.
- 64bit 133MHz PCI-X interface.
- Built-in Watchdog Timer (WDT) function.
- Easy configuration for by-pass mode via software programming.
- Built with both on-board LED for LAN status and bypass mode,

2.2 Installation

Please check the included items in the box before installation. If there is any missing part, please contact the local dealer. Including items:

1. **ABN-182** PCI-X Ethernet controller card
2. Driver CD
3. User's Guide

You can follow the step to install the Ethernet controller card to the system.

- Read this Manual
- Setup the configuration from software.
- Install the card on the available PCI-X slot.
- Prepare driver for the card, which can be found in the Driver CD or download update driver form Intel website.
- Connect the Ethernet cable to check the board function.

CAUTION: Make sure the card is PROPERLY installed on a PCI-X slot; reverse or shifted installation can cause serious damage on the Ethernet controller or your system.

Programming Setup

3.1 Programming Support

The bypass test program for ABN-182 supports Linux operating system: Fedora core/Red Hat 9.0. Test programs base on gcc version 4.1.0 20060304 (Fedora Core 5). And gcc version 3.2.2 20030222(red-hat 9) The following sections will cover Linux program installation and function on OS, for other supported, please contact Portwell.

3.2 Programming Configuration

The following steps are manual setup for Linux 2.4.X/2.6.X kernel:

- Copy ABN-182 bypass program to Linux OS
- Select OS kernel version (kernel 2.4 or 2.6) and go into path.
- Go into ap path to make test program:
  [root @ host]# cd ap
  [root @ host ap]# ./complie.sh
- Run test program and go into main test screen.
  [root @ host ap]# ./test_all

Portwell ABN182 Bypass program	version:1.00
Author: Angus Yang	
function select:	
[1] set to normal mode	[2] set to bypass mode
[3] set bypass enable function	[4] set Next boot enable function
[5] watch dog timer setting	
[ESC] exit program	
Select function:1	

1. [1]set to normal mode: set bypass mode disable.
2. [2]set to bypass mode: set bypass mode enable.
3. [3]set bypass enable function: set bypass mode when the system power off

bypass enable function	
function select:	
[1] set to bypass enable	[2] set to bypass disable
[ESC] return main window	
Select function:1	

4. [4] set next boot enable function: set bypass mode disable /enable when system boot next time.

Next boot enable fucntion	
function select:	
[1] set next boot enable	[2] set next boot disable
[ESC] return main window	
Select function:1	

5. [5] watch dog timer setting: bypass function enabled and disable base on watch dog setting.
   - Select [1] to clear watch dog timer old setting. Reset the system to normal mode.
   - Select the watch dog timer you want to test: [4]~[e]
   - Select [3] to start watch dog timer function. The bypass mode will enabled when time is up.

watch dog timer setting	
function select:	
[1] clear watch dog timer and set it to normal mode	
[2] disable bypass&watch dog timer	
[3] refresh bypass&watch dog timer	
[4] set wdt cycle:264.528 ms	[5] set wdt cycle:528.516 ms
[6] set wdt cycle:1057.032 ms	[7] set wdt cycle:2214.065 ms
[8] set wdt cycle:4228.129 ms	[9] set wdt cycle:8456.258 ms
[a] set wdt cycle:16912.52 ms	[b] set wdt cycle:33825.03 ms
[c] set wdt cycle:67550.06 ms	[d] set wdt cycle:135300.1 ms
[e] set wdt cycle:270600.3 ms	
[ESC] return main window	
Select function:	

*Note: all functions are for single card test, it does not support more cards test at the same time this moment.

3.3 Programming Guide

Following is API description.

1 dev_found

Function name	dev_found
Description	To find the NIC device
Format:	
int dev_found(unsigned char prod_num) //v1.04 Input: prod_num 1-1. PROD_ABN182: bypass card of abn182 Return: 0: device was found successfully -1:can not find device.	

2. set_to_normal

Function name	set_to_normal
Description	Set Ethernet segment to normal mode
Format:	
int set_to_normal (unsigned char seg, unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do set to normal function. 2. proc_dev: 2-1. PROD_ABN182: bypass card of abn182 Return: 0: return ok. -1:return fail.	

3. set_to_nonormal

Function name	set_to_nonormal
Description	Set Ethernet segment to non-normal mode
Format:	
int set_to_nonormal(unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do set_to_nonormal function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 0: return ok. -1:return fail.	

4. set_nextboot

Function name	set_nextboot
Description	1. Set Ethernet segment to normal mode or non-normal mode when boot up next time
Format:	
int set_nextboot (unsigned char mode,unsigned char seg,unsigned char proc_dev)//v1.04 Input: 1. mode: 1-1. SET_NB_NORMAL: set Ethernet segment to normal mode 1-2. SET_NB_NO_NORMAL: set Ethernet segment to non-normal mode. 2. seg:It assign which card in structure of card type to do set_nextboot function. 3. proc_dev: 3-1. PROD_ABN182 :bypass card of abn182 Return: 0: return ok. -1:return fail.	

5. set_bpe

Function name	set_bpe
Description	1. Set Ethernet segment to open mode or bypass mode when power fails or watch dog timer expires
Format:	
int set_bpe (unsigned char mode,unsigned char seg,unsigned char proc_dev)//v1.04 Input: 1. mode: SET_BPE_OPEN: set Ethernet port to open mode. SET_BPE_BYPASS: set Ethernet port to bypass mode. 2. seg:It assign which card in structure of card type to do set_bpe function. 3. proc_dev: 3-1. PROD_ABN182 :bypass card of abn182 Return: 0: return ok. -1:return fail.	

6 set_period_wdt

Function name	set_period_wdt
Description	Set watch dog timer period
Format:	

int set_period_wdt (unsigned char pd,unsigned char seg,unsigned char proc_dev)//v1.04

Input:

1. pd

SET_WDT_PERIOD_264258:set watch dog timer period as 264.258ms

SET_WDT_PERIOD_528516:set watch dog timer period as 528.516ms

SET_WDT_PERIOD_1057032:set watch dog timer period as 1057.032ms

SET_WDT_PERIOD_2114065:set watch dog timer period as 2114.065ms

SET_WDT_PERIOD_4228129:set watch dog timer period as 4228.129ms

SET_WDT_PERIOD_8456258:set watch dog timer period as 8456.258ms

SET_WDT_PERIOD_1691252:set watch dog timer period as 16912.52ms

SET_WDT_PERIOD_3382503:set watch dog timer period as 33825.03ms

SET_WDT_PERIOD_6755006:set watch dog timer period as 67550.06ms

SET_WDT_PERIOD_1353001:set watch dog timer period as 135300.1ms

SET_WDT_PERIOD_2706003:set watch dog timer period as 270600.3ms

2.seg:It assign which card in structure of card type to do set_period_wdt function.

3.proc_dev:

3-1. PROD_ABN182: bypass card of abn182

Return: 0: return ok.

-1:return fail.

7. arm_wd

Function name	arm_wdt
Description	Test mode changing rate by watchdog timer, and its period is set by set_period_wdt function (6-6). Its mode will be changed to bypass-mode when watchdog timer time out.
Format:	
int arm_wd (unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do arm _wdt function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 0: return ok. -1:return fail.	

8. dis_arm_wdt_to_sts

Function name	Dis_arm_wdt_to_sts
Description	Clear the set of watchdog timer.
Format:	
int dis_arm_wdt_to_sts(unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do arm _wdt function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of ab n182 Return: 0: return ok. -1:return fail.	

9. dis_bp_wdt

Function name	dis_bp_wdt
Description	Disable watch dog timer and keep status as current state
Format:	
1. If Ethernet is on bypass mode, when user does this command, it will be disable watch dog and keep it in bypass mode. If normal mode, after doing this command, it will be kept in normal mode. int dis_bp_wdt(unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do arm _wdt function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 0: return ok. -1:return fail.	

10 read_status_now

Function name	read_status_now
Description	Read bypass status
Format:	
unsigned char read_status_now(unsigned char seg,unsi gned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do read_status_now function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 0: normal mode 1: open mode 2: bypass mode	

11 read_settint_wdt

Function name	read_setting_wdt
Description	Read watch dog timer setting
Format:	
unsigned char read_setting_wdt(unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do read_setting_wdt function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: Bit1=0: watch dog timer dis-armed. Bit1=1: watch dog timer armed. Bit0=0: watch dog timer did not expire Bit0=1: watch dog timer expire	

12 read_mode_nb

Function name	read_mode_nb
Description	Read next boot status
Format:	
unsigned char read_mode_nb(unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do read_mode_nb function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 0: next boot=normal mode. 1: next boot=non-normal mode & BPE=open mode. 2: next boot=non-normal mode & BPE=bypass mode.	

13 read_mode_pfwdte

Function name	read_mode_pfwdte
Description	Read power fails & watch dog timer expire status
Format:	
unsigned char read_mode_pfwdte (unsigned char seg,unsigned char proc_dev) //v1.04 Input: 1. seg:It assign which card in structure of card type to do read_mode_pfwdte function. 2. proc_dev: 2-1. PROD_ABN182 :bypass card of abn182 Return: 1: pfwdte to open mode. 2: pfwdte to bypass mode.	

4

Driver Installation

4.1 Driver Support

The drivers for **ABN-182** support the following OS Windows 2000, XP, NT, 98/98SE, ME, 95, Linux Kernel 2.2.x/2.4.x. The following sections will cover driver installation on some OS, for other supported drivers, please contact Portwell or refer to Intel

4.2 Driver Setup for Windows® 2000

The following steps are manual installation for Windows 2000

- Boot Windows 2000 then insert Driver CD to CD-ROM.
- Double-click pro2kxp.exe in ***\Pro1000\Win\Win2KXP***, and then An Wizard window will pop up click ***Next>***.
- Click ***NEXT>*** button, the Wizard summary window will appear.
- Click ***Finish*** button, then click ***CLOSE*** button.

4.3 Driver Setup for Windows® NT

The following steps are manual installation for Windows NT

- Boot Windows NT then insert Driver CD to CD-ROM.
- Double-click pront4.exe in ***\Pro1000\Win\Winnt4*** , and then An Wizard window will pop up click ***Next>***.
- Click ***NEXT>*** button, the Wizard summary window will appear.
- Click ***Finish*** button, then click ***CLOSE*** button.
- Reboot the system.

4.4 Driver Installation for Linux

Some of the kernel has the driver for ABN-182, and it will automatically add the new hardware when booting up. If the driver is not included in the kernel, please following the step to install.

- Get the driver from CD or download from Intel website.

NOTE: For ABN-182, the driver is **e1000-Driver_Version.tar.gz**

- Make the file and install to the machine

  Example: Installation of e1000-7.1.19.tar.gz on Fedora core 4.

  ```
  [root @ host~]# tar zxvpf e1000-7.1.19.tar.gz
  [root @ host~]# cd e1000-7.1.19/src/
  [root @ host src]# make install
  [root @ host src]# modprobe e1000
  ```

NOTE: Please consult with your vender or read the reference document about compiling driver for different version of kernel.

- Reboot the system if necessary.